EXPRESS MAIL NO. ER399790061US

PATENT
Attorney Docket No. 02-4120

UNITED STATES PATENT APPLICATION

OF

Brig Barnum ELLIOTT

FOR

SYSTEMS AND METHODS FOR FORWARDING DATA IN A COMMUNICATIONS NETWORK

SYSTEMS AND METHODS FOR FORWARDING DATA IN COMMUNICATIONS NETWORKS

FIELD OF THE INVENTION

[0001] The present invention relates generally to communications networks and, more particularly, to systems and methods for forwarding data in communications networks.

BACKGROUND OF THE INVENTION

[0002] Communications networks have existed for decades. To route datagrams, such as packets, cells, etc., through such networks, each routing node in the networks may rely on forwarding tables. Forwarding tables provide routing nodes with instructions for forwarding a received datagram "one hop" further towards its destination. Specifically, a routing node can inspect one or more fields in a datagram, look up a corresponding entry in a forwarding table, and then put the datagram on the indicated queue for outbound transmission across one of a group of outbound links associated with the routing node.

[0003] Many variants exist on how to build and use forwarding tables. Many existing routing nodes, for instance, create multiple forwarding tables. Then, when a datagram arrives, the routing node selects which forwarding table should be used for this particular datagram. The choice of forwarding table typically depends on the protocol used by the datagram, the datagram's relative priority, or the like.

[0004] When failures occur in the network (e.g., a link fails), the routing nodes may often need to be supplied with (or need to generate) new forwarding tables to allow the nodes to route datagrams around the failures, which can cause delays in the network. Therefore, there exists a need to improve data routing in communications networks.

SUMMARY OF THE INVENTION

[0005] Systems and methods consistent with the principles of the invention provide improved techniques for routing data in a communications network.

[0006] In accordance with an exemplary implementation consistent with the principles of the invention, a communications network includes at least one control station and a group of network nodes. The at least one control station generates batches of forwarding tables, where each batch of forwarding tables includes a primary forwarding table and a group of backup forwarding tables, and forwards the batches of forwarding tables. Each of the network nodes is associated with one or more outbound and inbound links and is configured to receive a batch of forwarding tables from the at least one control station and install the primary forwarding table from the batch as a current forwarding table. Each network node is further configured to detect that a quality of one of an outbound and inbound link has changed, generate a message instructing other nodes of the group of network nodes to switch to a backup forwarding table in response to detecting the quality change, and transmit the message to the other nodes.

[0007] In another implementation consistent with the principles of the invention, a control station in a communications network that includes a group of nodes, includes a processor and a memory configured to store topology information for the communications network. The processor is configured to generate a batch of forwarding tables for each of the group of nodes

based on the topology information, where each batch of forwarding tables includes a primary forwarding table and a group of backup forwarding tables, and cause each batch of forwarding tables to be transmitted to the corresponding node of the group of nodes.

[0008] In yet another implementation consistent with the principles of the invention, a method for routing data in a communications network is provided. The method includes receiving a group of forwarding tables, including a primary forwarding table and a group of backup forwarding tables, from a remote device; using the primary forwarding table as a current forwarding table for routing data in the communications network; and storing the group of backup forwarding tables. The group of backup forwarding tables enables continued routing of data in the communications network when at least one event occurs in the communications network.

[0009] In still another implementation consistent with the principles of the invention, a node, associated with at least one outbound link and at least one inbound link, that transmits data in the communications network is provided. The node includes a processor and a memory configured to store a primary forwarding table and a group of backup forwarding tables. The processor detects a change in quality in one of the at least one outbound link and the at least one inbound link, generates a message that identifies the detected one outbound link or inbound link, and causes the message to be transmitted to one or more other nodes in the communications network. The message instructs the one or more other nodes to switch to a backup forwarding table associated with the identified one outbound link or inbound link.

BRIEF DESCRIPTION OF THE DRAWINGS

[0010] The accompanying drawings, which are incorporated in and constitute a part of this specification, illustrate an embodiment of the invention and, together with the description, explain the invention. In the drawings,

[0011] Fig. 1 illustrates an exemplary communications network in which systems and methods consistent with the principles of the invention may be implemented;

[0012] Fig. 2 illustrates an exemplary hardware diagram of one of the backbone satellites of Fig. 1;

[0013] Fig. 3 illustrates an exemplary configuration of a batch of forwarding tables, in an implementation consistent with the principles of the invention, that may be relied upon by a forwarding engine in routing information in the network of Fig. 1;

[0014] Fig. 4 illustrates an exemplary configuration of the ground station of Fig. 1 in an implementation consistent with the principles of the invention;

[0015] Fig. 5 illustrates an exemplary configuration of the control station of Fig. 1 in an implementation consistent with the principles of the invention;

[0016] Fig. 6 illustrates an exemplary process for transmitting and processing batch messages in an implementation consistent with the principles of the invention;

[0017] Fig. 7 illustrates an exemplary configuration of a batch message in an implementation consistent with the principles of the invention;

[0018] Fig. 8 illustrates an exemplary process, performed by a network node, when a failure is detected by the network node in an implementation consistent with the principles of the invention;

[0019] Fig. 9 illustrates an exemplary configuration of a switch forwarding table message in an implementation consistent with the principles of the invention; and

[0020] Fig. 10 illustrates an exemplary process, performed by a network node, in response to receiving a switch forwarding table message in an implementation consistent with the principles of the invention.

DETAILED DESCRIPTION

[0021] The following detailed description of implementations consistent with the present invention refers to the accompanying drawings. The same reference numbers in different drawings may identify the same or similar elements. Also, the following detailed description does not limit the invention. Instead, the scope of the invention is defined by the appended claims and equivalents.

[0022] Implementations consistent with the principles of the invention ensure continued operation of a communications network when one or more events (e.g., link failure or degradation) occur in the communications network. To ensure the continued operation of the network, each node in the network stores a group of backup forwarding tables. When an event occurs, the nodes may switch to one of the backup forwarding tables and continue to route data based on the backup forwarding table.

EXEMPLARY NETWORK

[0023] Fig. 1 illustrates an exemplary communications network 100 in which systems and methods consistent with the principles of the invention may be implemented. In Fig. 1, curved line 101 represents the edge of the Earth. Network 100 may include a number of elements that correspond to either Earth-based or space-based network devices.

[0024] As shown in Fig. 1, the space-based network devices may include a number of "backbone" satellites 110-1, 110-2, and 110-3 (collectively backbone satellites 110), and a number of "user" satellites 120-1 through 120-4 (collectively user satellites 120) that communicate through backbone satellites 110 to obtain network service. Although three backbone satellites 110 and four user satellites 120 are shown in Fig. 1, one of ordinary skill in the art will recognize that these numbers are for illustrative purposes only.

[0025] Backbone satellites 110 communicate with each other over inter-satellite links, labeled as links 111 in Fig. 1. These links may be implemented as optical (e.g., laser) communication links or as conventional radio frequency (RF) links (e.g., microwave). In either case, backbone satellites 110 may use directional transmitters and receivers to communicate with one another. Directional transmitters and receivers allow backbone satellites 110 to communicate over longer distances than with omni-directional communication schemes. Directional communication schemes, however, require that each backbone satellite 110 know its location relative to another backbone satellite with which it would like to communicate so that it can point its transmitter/receiver in the correct direction.

[0026] Inter-satellite links 111 may be high capacity links. For example, when implemented using RF technology, they may run at 100s of megabits/second. When implemented with optical transmitters and receivers, they may run at 10s of gigabits/second.

[0027] User satellites 120 may communicate with backbone satellites 110 through access links 112 (shown in Fig. 1 as dotted lines). Access links 112 may be RF links than tend to be of lower capacity and have shorter ranges than inter-satellite links 111. Access links 112 may also be designed to only require omni-directional antennas on user satellites 120. Omni-directional antennas do not require the sophisticated pointing and tracking mechanisms that are required of directional antennas. Backbone satellites 110, in forming access links 112, may use, for example, omni-directional, patch, sectorized, or dish antennas. The particular antenna to use may depend on the specific services that are required. The RF communications forming access links 112 may be in a number of frequency bands, such as UHF band, L band, cellular (or GSM or PCS) bands, ISM bands (910 MHz or 2.4 GHz), or other convenient frequency bands.

[0028] Network 100, in addition to including backbone satellites 110 and user satellites 120, may also include earth-based entities 130. As shown in Fig. 1, earth-based entities 130 (illustratively shown as a helicopter and a truck) may interface with network 100 through access links 112 in a manner similar to user satellites 120.

[0029] Backbone satellites 110 may connect to one or more ground stations 140-1 through 140-3 (collectively ground stations 140) via up/down links 113. Up/down links 113 may include high capacity links designed for communication between a satellite and a ground terminal.

Ground stations 140 may include fairly large and established ground terminals that have high capacity links designed for communications with satellites. Ground stations 140 may include, for example, large dish antennas that communicate through an RF connection with backbone satellites 110. The RF connection may run at, for example, 1 gigabit/second.

[0030] Ground stations 140 may connect to one another through standard terrestrial links, such as fiber optic links 114. One of ordinary skill in the art will appreciate that other types of terrestrial links, such as, for instance, coaxial cable and freespace optical connections are also possible.

[0031] Ground stations 140 may also act as network gateways to other private or public networks, such as network 150. In the case of a public network, network 150 may be the Internet. In the case of a private network, network 150 may be, for example, a proprietary military or corporate network. In some cases, network 150 may include a private portion and a public portion. In general, networks 100 and 150 allow any entity that can connect to network 150 the ability to communicate through the satellite portion of network 100.

[0032] Control stations 160-1 and 160-2 (collectively control stations 160) store network topology information (and other information) for controlling the forwarding of information throughout network 100. While two standalone control stations 160 are illustrated in Fig. 1, one skilled in the art will appreciate that a typical network may include more or fewer control stations. Moreover, a control station 160 may be implemented in one or more of the other devices illustrated in Fig. 1. For example, in one implementation, a control station 160 may be implemented in a backbone satellite 110. In exemplary implementations consistent with the principles of the invention, one control station 160 acts as a primary station, while the other control stations 160 may act as backup stations, thereby providing system redundancy and robustness. All control stations 160 may store the same set of location information and satellite descriptions, execute the same algorithms, and hence, all should derive the same topology information. However, only the primary station 160 may disseminate new topology commands to background satellites 110 and ground stations 140.

[0033] Network 100 may transmit data using a packet-based transmission protocol, such as the well known Internet Protocol (IP). Under the IP protocol, each device in network 100 is associated with an IP address that uniquely identifies it from all other devices. Data sent under the IP protocol is broken into data units called packets, each of which includes the IP address that identifies the destination for the packet. A packet "hops" from one device to another in network 100 until it is received by its destination device.

SATELLITE ARCHITECTURE

[0034] Fig. 2 illustrates an exemplary hardware diagram of one of backbone satellites 110 of Fig. 1. Backbone satellite 110, as well as being a communication satellite, may act as a router in network 100. Backbone satellite 110 may include a redundant implementation to facilitate fault tolerance. In Fig. 2, this is shown as an "A" side architecture and a "B" side architecture.

[0035] The A side architecture may include a read-only-memory (ROM) 201, a processor (CPU) 202, random access memory (RAM) 203, and forwarding engine (FWD) 204. A cross-bar bus (X-BAR) 205 may connect RAM 203 and forwarding engine 204 to input/output components 221-224.

[0036] The B side architecture may be implemented in an essentially identical manner to the A side architecture and acts as a backup in case of a failure in the A side architecture. In particular, the B side architecture may include ROM 211, a CPU 212, RAM 213, and forwarding engine 214, which utilizes cross-bar 215.

[0037] ROM 201 and 211 may each contain all necessary read-only storage for backbone satellite 110. ROM 201 and 211 may, for example, store programming instructions for operation of the backbone satellite, geo-locations of some or all ground stations, system identifiers, configuration parameters, etc. Although shown as single monolithic ROM devices, ROM 201 and 211 may be implemented as a mix of different types of non-volatile memory, and may even include a certain amount of reprogrammable memory as well. For instance, ROM 201 or 211 may be implemented as ROM, EEPROM, flash memory, etc.

[0038] CPUs 202 and 212 may be embedded processors that execute computer instructions. CPUs 202 and 212 may generally manage the control and routing functions for backbone satellite 110.

[0039] Forwarding engines 204 and 214 may each include high-speed data forwarding paths that obtain header information from packets received by backbone satellite 110, and based on the header information, may retransmit the packets on a link that leads towards the final destination of the packet. To increase processing speed, forwarding engines 204 and 214 may be implemented as FPGAs (field programmable gate arrays), ASICs (application specific integrated circuits), or other high-speed circuitry. In general, forwarding engines 204 and 214 implement many of the core routing functions of backbone satellite 110, and thus, in conjunction with their CPUs and RAMs, function as routers in the satellite. The design and implementation of routers and routing techniques is generally known in the art and will thus not be described further herein.

[0040] Forwarding engines 204 and 214 may include one or more forwarding tables that store information relating to packet forwarding. The forwarding tables may alternatively be stored in RAM 203 and 213. In one implementation, forwarding engines 204 and 214 store a "batch" of forwarding tables. This batch contains a whole set of fallback (or backup) forwarding tables for any unexpected event or sequence of events that can occur in network 100 (e.g., a link failing). Forwarding engines 204 and 214 may store forwarding tables based on the protocol used by backbone satellite 110. For example, forwarding engines 204 and 214 may use forwarding tables for IP, asynchronous transfer mode (ATM), Multi-Protocol Label Switching (MPLS), fast packet switching, Ethernet, and the like for routing data in network 100.

[0041] Fig. 3 illustrates an exemplary batch 300 of forwarding tables, in an implementation consistent with the principles of the invention, that may be relied upon by forwarding engine 204/214 in routing information in network 100. It will be appreciated that each node (e.g., backbone satellites 110, user satellites 120, ground stations 140, etc.) in network 100 may store a similar batch 300 of forwarding tables that enables that particular node to continue to route data when unexpected events occur in network 100.

[0042] As illustrated, batch 300 may include a primary forwarding table 305 and a group of backup forwarding tables 310. Forwarding engine 204/214 may rely on primary forwarding table 305 when no problems exist in network 100 (i.e., no unexpected events have occurred). Forwarding engines 204 and 214 may also store one or more backup forwarding tables 310, which may be used by forwarding engine 204/214 when one or more unexpected events occur in network 100. In one implementation consistent with the principles of the invention, the number of backup forwarding tables 310 for a given node may be *ns* x *nl* x *nq*, where *ns* represents the total number of nodes in network 100, *nl* represents the total number of outbound links (e.g., transmitters) associated with a given node, and *nq* represents the total number of different non-normative "qualities" that a given outbound link can take.

[0043] As an example, assume that network 100 includes 10 satellites 110/120 and 2 ground stations 140. In this situation, *ns* would equal 12. Moreover, assume, for example, that backbone satellite 110-2 (Fig. 1) has 4 laser transmitters and 1 up/down link. For this satellite 110-2, *nl* would equal 5. It will be appreciated that value *nl* may vary for other nodes in network 100 (i.e., the other nodes may include more or fewer transmitters and/or links). If an outbound link for satellite 110-2 can run in any of three different modes (e.g., normal, degraded, failed), then *nq* would equal 2 for that given link since two non-normative states exist for that link (i.e., degraded or failed). Based on the above exemplary numbers, satellite 110-2 would then store a batch 300 of

$$1 + (12 \text{ x } 5 \text{ x } 2) = 121$$

different forwarding tables – one primary forwarding table 305 and 120 backup forwarding tables 310.

[0044] Batch 300 may thus contain backup forwarding tables 310 for each possible single unexpected event that may occur in network 100. If all links are operating normally, then forwarding engine 204/214 may use primary forwarding table 305. If, for example, link 2 (L2) on a satellite S1, which could correspond to backbone satellite 110-1, unexpectedly goes into a degraded or failed mode, designated as Q2, then forwarding engine 204/214 may use the backup forwarding table #S1-L2-Q2 instead of primary forwarding table 305.

[0045] As indicated above, each node in network 100 would contain its own unique batch 300 of forwarding tables. Thus, in a network containing 12 nodes, as per the example above, there would be 12 distinct batches 300 of forwarding tables, one for each node in the network.

[0046] While the above example described the situation in which a node in network 100 includes backup forwarding tables 310 to ensure continued traffic routing when any single event occurs in network 100, implementations consistent with the present invention are not so limited. For example, in another implementation, each node in network 100 may include backup forwarding tables 310 for handling only a subset of all events that could occur in network 100 (e.g., the most probable single event failures). Alternatively, batch 300 could include enough backup forwarding tables 310 to handle any two-event scenario. In such situations, each node may include enough backup forwarding tables 310 to ensure continued operation when any two links in network 100 experience unexpected events (e.g., degraded modes, failure, etc.). In yet other implementations, each node in network 100 may contain enough backup forwarding tables 310 to handle any possible combination of unexpected events.

[0047] Returning to Fig. 2, RAM 203 and 213 include volatile memory in which data packets and/or other data structures may be stored and manipulated. I/O devices 221-224 may access RAM 203 and 213. RAM 203 and 213 may store queues of packets that can be read and transmitted by I/O devices 221-224.

[0048] I/O devices 221-224 contain the hardware interfaces, transceivers, and antennas (or telescopes) that implement links 111-113. ACC I/O device 211 handles access links 112. ISL I/O devices 222 and 223 handle inter-satellite links 111. UPD I/O device 224 handles up/down links 113.

[0049] Although backbone satellite 110 is shown as having four I/O devices 221-224, one of ordinary skill in the art will recognize that backbone satellite 110 could have more or fewer I/O devices. Further, multiple I/O devices, such as ISL I/O devices 222 and 223, may be operated in unison to form a single high capacity link.

GROUND STATION ARCHITECTURE

[0050] Fig. 4 illustrates an exemplary configuration of ground station 140 of Fig. 1 in an implementation consistent with the principles of the invention. It will be appreciated that the configuration illustrated in Fig. 4 is provided for explanatory purposes only and that many other configurations are possible.

[0051] As illustrated, ground station 140 may include a redundant implementation to facilitate fault tolerance. In Fig. 4, this is shown as an "A" side architecture and a "B" side architecture.

[0052] The A side architecture may include ROM 401, a processor (CPU) 402, RAM 403, and forwarding engine (FWD) 404. A cross-bar bus (X-BAR) 405 may connect RAM 403 and forwarding engine 404 to input/output components 420-422.

[0053] The B side architecture may be implemented in an essentially identical manner to the A side architecture and acts as a backup in case of a failure in the A side architecture. In particular, the B side architecture may include ROM 411, a CPU 412, RAM 413, and forwarding engine 414, which utilizes cross-bar 415.

[0054] ROM 401 and 411 may each contain all necessary read-only storage for ground station 140. ROM 401 and 411 may, for example, store programming instructions for operation of the ground station, geo-locations of some or all backbone satellites 110, system identifiers, configuration parameters, etc. Although shown as single monolithic ROM devices, ROM 401 and 411 may be implemented as a mix of different types of non-volatile memory, and may even include a certain amount of reprogrammable memory as well. For instance, ROM 401 or 411 may be implemented as ROM, EEPROM, flash memory, etc.

[0055] CPUs 402 and 412 may be embedded processors that execute computer instructions. CPUs 402 and 412 may generally manage the control and routing functions for ground station 140.

[0056] Forwarding engines 404 and 414 may each include high-speed data forwarding paths that obtain header information from packets received by ground station 140, and based on the header information, may retransmit the packets on a link that leads towards the final destination of the packet. To increase processing speed, forwarding engines 404 and 414 may be implemented as FPGAs, ASICs, or other high-speed circuitry. In general, forwarding engines 404 and 414 implement many of the core routing functions of ground station 140, and thus, in conjunction with their CPUs and RAMs, function as routers in the ground station. The design and implementation of routers and routing techniques is generally known in the art and will thus not be described further herein.

[0057] Forwarding engines 404 and 414 may include one or more forwarding tables that store information relating to packet forwarding. The forwarding tables may alternatively be stored in RAM 403 and 413. In one implementation, forwarding engines 404 and 414 store a unique batch 300 of forwarding tables 305 and 310 in a manner similar to that described above with respect to Fig. 3.

[0058] RAM 403 and 413 include volatile memory in which data packets and/or other data structures may be stored and manipulated. I/O devices 420-422 may access RAM 403 and 413. RAM 403 and 413 may store queues of packets that can be read and transmitted by I/O devices 420-422.

[0059] I/O devices 420-422 contain the hardware interfaces, transceivers, and antennas (or telescopes) that implement links 113 and 114. Inter-ground station links (IGSL) I/O devices 420 and 421 handle links 114. UPD I/O device 422 handles up/down links 113.

[0060] Although ground station 140 is shown as having three I/O devices 420-422, one of ordinary skill in the art will recognize that ground station 140 could have more or fewer I/O devices. Further, multiple I/O devices, such as IGSL I/O devices 420 and 421, may be operated in unison to form a single high capacity link.

CONTROL STATION ARCHITECTURE

[0061] Fig. 5 illustrates an exemplary configuration of one of control stations 160 of Fig. 1 in an implementation consistent with the principles of the invention. It will be appreciated that the configuration illustrated in Fig. 5 is provided for explanatory purposes only and that many other configurations are possible. Moreover, it will be appreciated that a typical control station 160 may include other components than those illustrated in Fig. 5 that aid in receiving, processing, and/or transmitting data.

[0062] As illustrated, control station 160 may include a bus 510, a processor 520, a memory 530, an input device 540, an output device 550, and a communication interface 560. Bus 510 permits communication among the components of control station 160.

[0063] Processor 520 may include any type of conventional processor or microprocessor that interprets and executes instructions. Memory 530 may include a RAM or another type of dynamic storage device that stores information and instructions for execution by processor 520, a

ROM device and/or another type of static storage device that stores static information and instructions for processor 520, and/or a magnetic disk or optical disk and its corresponding drive.

[0064] Input device 540 may include one or more conventional mechanisms that permit an operator to input information to control station 160, such as a keyboard, pointing device (e.g., a mouse, a pen, or the like), one or more biometric mechanisms, such as a voice recognition device, etc. Output device 550 may include one or more conventional mechanisms that output information to the operator, such as a display, a printer, a speaker, etc. Communication interface 560 may include any transceiver-like mechanism that enables control station 160 to communicate with other devices and/or systems. For example, communication interface 560 may include a modem or an Ethernet interface to a network. Alternatively, communication interface 560 may include other mechanisms for communicating via a data network, such as network 150.

[0065] Control station 160 may implement the functions described below in response to processor 520 executing software instructions contained in a computer-readable medium, such as memory 530. A computer-readable medium may be defined as one or more memory devices and/or carrier waves. In alternative embodiments, hardwired circuitry may be used in place of or in combination with software instructions to implement features consistent with the principles of the invention. Thus, implementations consistent with the principles of the invention are not limited to any specific combination of hardware circuitry and software.

EXEMPLARY PROCESSING

[0066] Fig. 6 illustrates an exemplary process for transmitting and processing batch messages in an implementation consistent with the principles of the invention. In general, batch messages are used to forward primary forwarding tables and backup forwarding tables to nodes in network 100. Processing may begin with a control station, such as control station 160 (Fig. 1), generating a batch message for a node, such as a backbone satellite 110 or ground station 140, in network 100 (act 605). As set forth above, control station 160 may store the topology of the entire network 100 and may generate the batch message based on the network topology. Control station 160 may generate the batch message when network topology is going to change or any other time at which new forwarding tables are needed for network 100 (e.g., after central station 160 determines that an unexpected event has occurred, when a new node is added to network 100, etc.).

[0067] The batch message may include a primary forwarding table 305 and a group of backup forwarding tables 310. As described above, the number of backup forwarding tables 310 may be enough to handle any single unexpected event, a subgroup of possible unexpected events, or any combination of two or more unexpected events.

[0068] Fig. 7 illustrates an exemplary configuration of a batch message 700 in an implementation consistent with the principles of the invention. As illustrated, batch message 700 may include a protocol field 710, a destination node identifier (ID) field 720, and a batch field 730. It will be appreciated that batch message 700 may include other fields than those illustrated in Fig. 7. For example, batch message 700 may include a field identifying the control station from which batch message 700 is transmitted.

[0069] Protocol field 710 may include information identifying message 700. In one implementation, protocol field 710 may store information indicating that message 700 is a new batch message 710. Destination node ID field 720 may store information identifying the particular node to which batch message 700 is destined. Batch field 730 may store batch 300 of forwarding tables for the node identified in destination node ID field 720.

[0070] Once generated, control station 160 may transmit batch message 700 to the destination node (act 610). Control station 160 may generate and transmit batch messages 700 in the above-described manner for all of the nodes in network 100. Control station 160 may implement the following acts, expressed as a high-level pseudo-code, to ensure that the requisite number of forwarding tables (primary 305 and backup 310) are generated for each node in network 100:

```
For I = 1 to $ns$
    Compute Primary Forwarding Table for node I
    For J = 1 to $ns$
        For K = 1 to $nl$
            For L = 1 to $nq$
                Compute Backup Forwarding Table #J-K-L for node $ns$,
                and add to batch
            Next L
        Next K
    Next J
    Store batch in batch message
    Disseminate batch to node I
Next I.
```

[0071] The network node may receive batch message 700 from control station 160 (act 615). In response, the network node may authenticate and validate the contents of batch message 700 (act 620). To authenticate and validate batch message 700, the network node may, for example, check destination node field 720 to ensure that batch message 700 is intended for this particular network node. The network node may also check that batch message 700 includes the correct number of primary and backup forwarding tables.

[0072] The network node may install primary forwarding table 305 from batch message 700 as the current forwarding table (act 625). Network node may also replace any existing backup forwarding tables with backup forwarding tables 310 from batch message 700 (act 625).

[0073] If a given network node detects a local failure, the network node may generate a command telling other network nodes to switch to the appropriate backup forwarding table 310 instead of primary forwarding table 305. In general, the network node may detect a local failure in a number of different ways (e.g., the network node’s transmitter equipment may be able to generate a signal that it has failed, a "loss of carrier" indication may be associated with the link, "hello packets" may cease flowing, etc.). Any of these techniques may be used to deduce that an outbound link has failed, and to trigger the transmission of a switch message.

[0074] Fig. 8 illustrates an exemplary process, performed by a network node (e.g., a backbone satellite 110, a ground station 140, etc.), when a failure is detected by the network node. It is assumed, for explanatory purposes, that the network node is node #I. Processing may begin with the network node detecting a quality degradation (failure, degraded mode, etc.) of an outbound (or inbound) transmission link #K, associated with the network node (act 805). The quality of this link as a result of the failure (or quality degradation) will be denoted as #L. The network node may generate a switch forwarding table message to indicate to other nodes in network 100 that they are to switch to backup forwarding table #I-K-L (act 810). The remainder of Fig. 8 is discussed below.

[0075] Fig. 9 illustrates an exemplary configuration of a switch forwarding table message 900 in an implementation consistent with the principles of the invention. As illustrated, switch forwarding table message 900 may include a protocol field 910, an originating node ID field 920, a link ID field 930, and a new quality for link field 940. It will be appreciated that switching forwarding table message 900 may include other fields than those illustrated in Fig. 9. For example, switch forwarding table message 900 may include a field identifying whether the link in link ID field 930 is an outbound or inbound link.

[0076] Protocol field 910 may include information identifying message 900. In one implementation, protocol field 910 may store information indicating that message 900 is a message instructing other network nodes to switch forwarding tables. Originating node ID field 920 may store information identifying the particular node from which switch forwarding table message 900 originated. In the example above, originating node ID field 920 may store information identifying the network node as node #I. Link ID field 930 may store information identifying the link associated with the network node identified in originating node ID field 920 that has experienced the event (e.g., failure, degradation, etc.). In the example above, link ID field 930 may store information identifying the link as #K. New quality for link field 940 may store information identifying the quality of the link identified in link ID field 930. In the example above, new quality for link field 940 may store information identifying the quality as #L.

[0077] Once switch forwarding table message 900 is generated, the network node may transmit switch forwarding table message 900 to the other network nodes in network 100 (act 815). The network node may also transmit switch forwarding message 900 to all control stations 160 in network 100 (act 820).

[0078] Fig. 10 illustrates an exemplary process, performed by a network node (e.g., a backbone satellite 110, a ground station 140, etc.), in response to receiving a switch forwarding table message 900 in an implementation consistent with the principles of the invention. Processing may begin with the network node receiving a switch forwarding table message 900 (act 1005). In response, the network node may authenticate and validate switch forwarding table message 900 (act 1010). To authenticate and validate switch forwarding table message 900, the network node may, for example, verify that the backup forwarding table identified in switch forwarding table message 900 exists.

[0079] The network node may determine if it is already using a backup forwarding table 310 from its stored batch 300 of forwarding tables (act 1015). If the network node is already using a backup forwarding table 310, processing may end. On the other hand, if the network node is using its primary forwarding table 305, the network node may switch to backup forwarding table 305 identified in switch forwarding table message 900 (act 1020). For example, if switch forwarding message 900 identifies the originating node as node #I, the link as link # K, and the new quality of link #K as #L, the network node may switch to backup forwarding table #I-K-L.

[0080] The network node may disseminate received switch forwarding table message 900 to other nodes in network 100 to implement this change throughout network 100 (act 1025). In general, the dissemination can be to some subset of the nodes in network 100, or to all nodes. In one implementation, a reliable flooding technique may be used for disseminating switch forwarding table message 900 throughout network 100. In some implementations consistent with the principles of the invention, a network node may piggy-back switch forwarding table message 900 with datagrams (or other data units, such as packets, cells, etc.) scheduled to be forwarded to other nodes. In this way, other nodes in network 100 will learn of the change in forwarding tables just as quickly as the nodes receive the datagrams being forwarded along the new path, and can adjust their forwarding tables immediately before even processing the datagram. Thus, the network node may immediately start forwarding datagrams along the revised path, without any time lag whatsoever. In essence, the datagrams themselves can carry the information needed to find the new path after a fault has occurred.

CONCLUSION

[0081] Implementations consistent with the principles of the invention ensure continued operation of a communications network when one or more events (e.g., link failure or degradation) occur in the communications network. To ensure the continued operation of the network, each node in the network stores a group of backup forwarding tables. When an event occurs, the nodes may switch to one of the backup forwarding tables and continue to route data based on the backup forwarding table.

[0082] The foregoing description of exemplary embodiments of the present invention provides illustration and description, but is not intended to be exhaustive or to limit the invention to the precise form disclosed. Modifications and variations are possible in light of the above teachings or may be acquired from practice of the invention. For example, while the above description focused on a satellite-based network, implementations consistent with the principles of the invention are not so limited. In fact, implementations consistent with the principles of the invention are equally applicable to terrestrial networks where continued routing of data upon the occurrence of events is desired.

[0083] While series of acts have been described with regard to Figs. 6, 8, and 10, the order of the acts may be varied in other implementations consistent with the present invention. Moreover, non-dependent acts may be implemented in parallel.

[0084] No element, act, or instruction used in the description of the present application should be construed as critical or essential to the invention unless explicitly described as such. Also, as used herein, the article "a" is intended to include one or more items. Where only one item is intended, the term "one" or similar language is used.

[0085] The scope of the invention is defined by the claims and their equivalents.